100 SQL interview questions

Only coding challenges | Reset progress | Search...

SQL Fundamentals

1. What is **SQL** and what is it used for? Question
2. Describe the difference between **SQL** and **NoSQL** databases. Question
3. What are the different types of **SQL commands**? Question
4. Explain the purpose of the **SELECT** statement. Question
5. What is the difference between **WHERE** and **HAVING** clauses? Question
6. Define what a **JOIN** is in SQL and list its types. Question
7. What is a **primary key** in a database? Question
8. Explain what a **foreign key** is and how it is used. Question
9. How can you prevent **SQL injections**? Question

Topic progress: 0%

11. Describe the concept of **denormalization** and when you would use it. Question

12. What are **indexes** and how can they improve query performance? Question

13. Explain the purpose of the **GROUP BY** clause. Question

14. What is a **subquery**, and when would you use one? Question

15. Describe the functions of the **ORDER BY** clause. Question

16. What are **aggregate functions** in SQL? Question

17. Explain the differences between **INNER JOIN**, **LEFT JOIN**, **RIGHT JOIN**, and **FULL JOIN**. Question

18. How do you insert a new row into a **database table**? Question

19. Explain how to update records in a **database table**. Question

20. What is a **SQL View** and what are its advantages? Question

SQL Data Types and Operators

21. List the different **data types** available in **SQL**. Question

22. What are the differences between **CHAR**, **VARCHAR**, and **TEXT** data types? Question

23. How do you use the **BETWEEN** operator in SQL? Question

24. Describe the use of the **IN** operator. Question

25. Explain the use of **wildcard characters** in SQL. Question

26. What is the purpose of the **LIKE** operator? Question

27. How do you handle **NULL values** in SQL? Question

28. What does the **COALESCE function** do? Question

29. What is the difference between **UNION** and **UNION ALL**? Question

30. Describe the use of **arithmetic operators** in SQL queries. Question

SQL Advanced Queries

31. Explain how to use the **CASE statement** in SQL. Question

32. How would you perform a **self JOIN**? Question

33. What is a **cross JOIN** and when would you use it? Question

34. How to implement **pagination** in SQL queries? Question

35. Explain the concept of **Common Table Expressions (CTEs)** and **recursive CTEs**. Question

36. What are **window functions** and how are they used? Question

37. How can you concatenate column values in SQL? Question

38. What is the **PIVOT operation** and how would you apply it? Question

39. Explain the process of combining a query that uses a **GROUP BY** with one that uses **ORDER BY**. Question

40. How would you find **duplicate records** in a table? Question

Database Design & Architecture

41. What is the **Entity-Relationship Model**? Question

42. Explain the different types of **database schema**. Question

43. What are **Stored Procedures** and how are they beneficial? Question

44. What is a **trigger** in SQL and when should it be used? Question

45. Describe the concept of **ACID** in databases. Question

Go up

Topic progress: 0%

47. How do **database indexes** work and what types are there? Question

48. Describe the process of **data warehousing**. Question

49. Explain the difference between **OLTP** and **OLAP** systems. Question

50. What are **materialized views** and how do they differ from standard views? Question

SQL Optimization and Performance

51. How do you identify and optimize **slow-running queries**? Question

52. What is **query execution plan** in SQL? Question

53. Explain how to use **EXPLAIN** or **EXPLAIN ANALYZE**. Question

54. How can **indexing** affect performance both positively and negatively? Question

55. Describe how to measure the performance of SQL queries. Question

56. How would you rewrite a query to improve its performance? Question

57. What are **partitioned tables** and how can they optimize performance? Question

SQL Security

58. How do you implement **database encryption** in SQL? Question

59. What are **roles** and how do they manage database access? Question

60. Explain the concept of **row-level security**. Question

61. Describe how to create and use **user-defined functions (UDFs)**. Question

SQL Functions and Expressions

62. Describe **scalar-valued** and **table-valued** functions. Question

63. How would you define a **stored procedure** with input and output parameters? Question

64. What is the difference between a **function** and a **stored procedure**? Question

65. How do you use the **CAST** and **CONVERT** functions? Question

Transaction Control and Locking

66. What is a **database transaction**? Question

67. Explain the concept of **locking** and its types in SQL databases. Question

68. What are the properties of **transactions**? Question

69. How do you manage **transaction isolation levels**? Question

70. What does it mean to **commit** or **roll back** a transaction? Question

SQL and Modern Data Ecosystems

71. How can SQL be integrated with **big data** technologies? Question

72. Discuss the interoperability of SQL with **cloud-based data stores**. Question

73. What is **Data Lake** and how can SQL interact with it? Question

74. Explain the interaction between SQL and **NoSQL** within the same application. Question

75. How does SQL work within a **microservices architecture**? Question

SQL Best Practices and Standards

76. What are some common SQL coding practices you follow? Question

77. How can you ensure the portability of **SQL scripts** across different database systems? Question

Go up

Topic progress: 0%

79. What are the benefits of using **stored procedures** instead of embedding SQL queries in code? Question

80. How do you document **SQL code** effectively? Question

Analytical SQL Questions

81. How would you find the Nth **highest salary** from a table? Question

82. How do you count the number of occurrences of a specific value in a column? Question

83. How can you calculate **running totals** in SQL? Question

84. Explain how to reverse the contents of a column without using a **reverse** function. Question

85. What approach do you use for creating a **calendar table**, and what are its uses? Question

Data Manipulation and ETL

86. What is the process of **Extract, Transform, Load (ETL)**? Question

87. How do you import/export data from/to a flat file using SQL? Question

88. Explain the steps for a basic **ETL process** in a data warehousing environment. Question

89. How do you cleanse and format data using SQL queries? Question

90. What tools do you use for automating data import/export routines? Question

Domain-Specific SQL Scenarios

91. How would you model a **many-to-many relationship** in SQL? Question

92. Describe how to manage **hierarchical data** in SQL. Question

93. How would you approach writing SQL queries for a reporting application? Question

94. Explain how to handle **temporal data** and time zones in SQL. Question

95. How do you use SQL in financial applications for risk and portfolio analysis? Question

Troubleshooting and Debugging

96. What steps do you take to **troubleshoot a failed SQL query**? Question

97. How can you recover data from a corrupt SQL database? Question

98. What methods do you employ to ensure **data integrity**? Question

99. How do you decipher and resolve **deadlocks** in SQL? Question

Advanced Data Analysis in SQL

100. Explain how to use SQL for **predictive analysis** and **machine learning** purposes. Question